DGA-Based Botnet Detection Using DNS Traffic

Yong-lin Zhou[1], Qing-shan Li[2], Qidi Miao[3]*, and Kangbin Yim[4]
[1] Computer Emergency Response Team, Beijing 100029, China
zyl@cert.org.cn
[2] MoE Key Lab. of Network and Software Security Assurance of Peking University
Beijing 100871, China
liqs@infosec.pku.edu.cn
[3] Software college, Northeastern University, Shenyang 110819, China
qidi_miao@126.com
[4] Soonchunhyang University, Asan 336745, Republic of Korea
yim@sch.ac.kr

Abstract

In recent years, an increasing number of botnets use Domain Generation Algorithms (DGAs) to bypass botnet detection systems. DGAs, also referred as "domain fluxing", has been used since 2004 for botnet controllers, and now become an emerging trend for malware. It can dynamically and frequently generate a large number of random domain names which are used to prevent security systems from detecting and blocking. In this paper, we present a new technique to detect DGAs using DNS NXDomain traffic. Our insight is that every domain name in the domain group generated by one botnet using DGAs is often used for a short period of time, and has similar live time and query style. We look for this pattern in DNS NXDomain traffic to filter out algorithmically generated domains that DGA-based botnets generate. We implemented our protosystem and carry outexperiment at a pilot RDNS of an Internet operator. The results show that our method is of good effectiveness on detecting algorithmically generated domains used by botnet.

Keywords: Domain Generation Algorithms, Domain fluxing, Domain names, NXDOMAIN

1 Introduction

The Domain Name System (DNS) is a critical component of the Internet infrastructure, mainly used to translate domain name to IP address. Currently most network services and applications rely on DNS. The domain name system does not distinguish the services between normal and malicious.

Botnets are composed of lots of malware-compromised machines which can be controlled through a command and control(C&C) communication channel[3]. Using botnets, the attacker can implement lots of malicious activities like stealing private info, spamming, phishing, DDoS attack, etc. According to the white paper published by Arbor Networks, botnets became one of the most threats to current Internet.

To bypass detection and blocking, enhance self-survival ability, and prolong lifetime, many botnets use DNS to organize and control. Previous used techniques include Dynamic DNS and fast flux, but recent botnets such as Conficker[2], Kraken[4, 5], Torpig[6], Srizbi and Bobax, introduced Domain Generation Algorithms into their command-and-control module. Domain Generation Algorithms is a technique, used by botnet to generate a large set of domain names but merely a small subset being used.

Current detection of DGAs mainly focused on domain name alphanumeric characters. Because of the difference between different DGAs and easy to change domain generate algorithms, Botnet can

Journal of Internet Services and Information Security (JISIS), volume: 3, number: 3/4, pp. 116-123

*Corresponding author: No.11 Wenhualu Heping district, Shenyang, Liaoning Province, China, 110819, Tel: +86-15504022631 change and replace their DGAs to avoid detection. But the access to domain names from domain-flux botnet has strong stability and regularity, not changing with domain name generation algorithms. So this simultaneous access can result in the same traffic including NXDomain traffic. In this paper, we propose a method for DGA-botnet detection. Based on the collected DNS NXDomain traffic at pilot RDNSs, we extract the active time and live span of each domain. Then we group domain names by domain level and parsed IP, and calculate domain access similarity for each group to get suspicious DGA-domain name list.

We apply our methodology to the DNS data set collected from several RDNSs and get some DGA-domains. The rest of this paper is organized as follows. In Section 2, we give related work. In Section 3, we present an overview of our detection methodology and describe DGA detection process in section 4. Next, the experimental results are presented in Section 5. Finally, in Section 6 we conclude.

2 Related Work

In the past, reverse engineering of botnet executable was often used. Based on reverse engineering, we can accurately find out how domain generation algorithm works, and then we can block the domain name generated before it is used.The first report on DGAs was Stone-Gross et.al[6]. Based on the reverse analysis of captured sample, Brett Stone-Grosset al.,who came from University of California, Santa Barbara, get the algorithm used by torpig. They found that each bot periodically generates a list of domains and then contacts it to locate active C&C servers.By utilizing fake C&C servers, and registering domain name ahead, they have successfully got the control of tropig and controlled it for ten days. Reverse engineering usually takes much time before the domain generation algorithm is cracked, and we need to get a related malware sample first.

The researchers from Texas A&M University and Narus.com developed a methodology to detect domain fluxing in DNS traffic. Based on observation they found that algorithmically generated domain names exhibit characteristics vastly different from legitimate domain names.Several distance metrics, including KL-distance,Edit distance and Jaccard measure are used to look at the distribution of alphanumeric characters[7].

Antonakakiset. al. From Damballa present a technique to detect randomly generated domains using Non-Existent Domain. There insight is that bots from the same botnet(with the same DGA algorithm) would generate similar NXDomain traffic[1].They uses a combination of clustering and classification algorithms, and there classification algorithm can assign generated domain clusters to models of known DGAs.

3 System Overview

In this section, we provide a high-level overview of our detection system. As shown in Figure 1, our system consists of two main modules: a Data collection and pre-handle module and a DGA Detection module. We describe these modules below in more detail.

There are a variety of data sets used in our system. First, in data collection module, we obtain a set of legitimate domain names from the top 10,000 most popular domains according to Alexa (alexa.com). Then, this module captures all NXDomain traffic from RDNS located in IDC. We get rid of DNS flows that including domains got from Alexa to decrease the DNS traffic. All data from RDNSs were merged for later use.

The DGA detection module analyzes the DNS traffic from data collection module (see Figure 1). We cluster all domains to group according to second TLD and parsed IP. Then, the clustered domains are calculated to see if the domains in one cluster have similar live time span and similar visit pattern (the

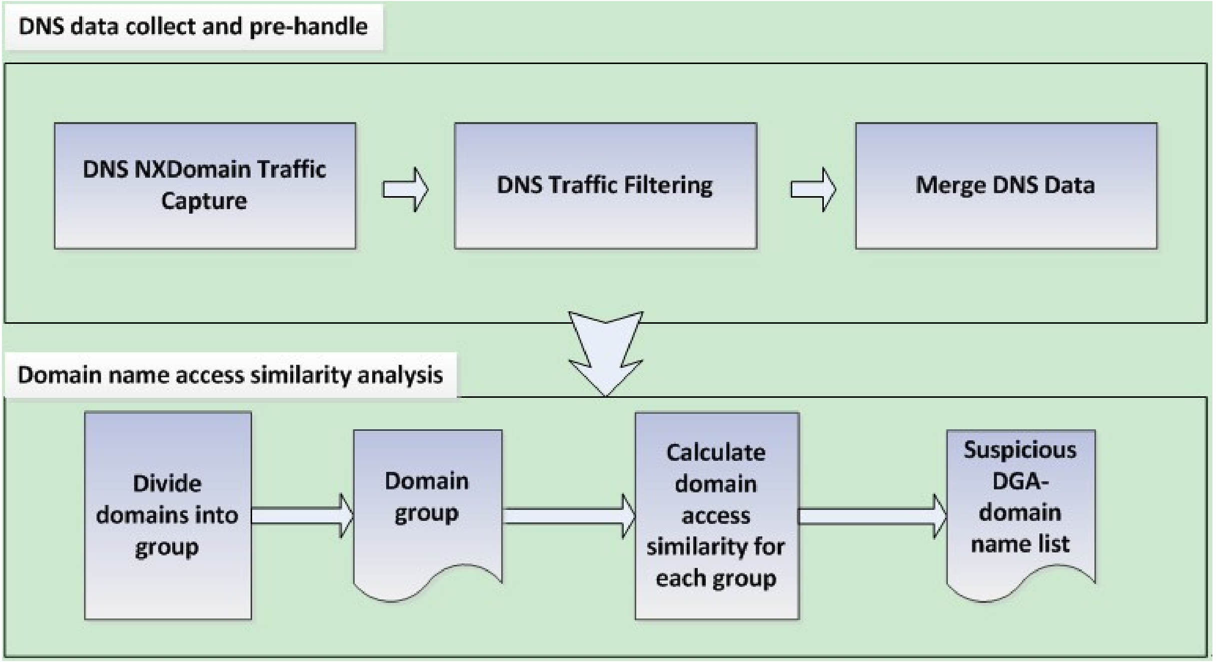
DNS data collect and pre-handle
DNS NXDomain Traffic Capture
DNS Traffic Filtering
Merge DNS Data
Domain name access similarity analysis
Divide domains into group
Domain group
Calculate domain access similarity for each group
Suspicious DGA-domain name list

Figure 1: overview of system

domains have been queried at one time simultaneously). The main objective of this module is to detect domain groups that likely use DGAs.

4 DGA Detection

In this section, we present our detection methodology that is based on calculating the similarity of live time span and visit pattern.

The detection method proposed in this paper is based on the observation that DGAs generated domains differ significantly from human generated domains in terms of the subdomain number, domain live life span and domain visit pattern. When a domain name is queried by a host, the domain name at this time is active. If there is a response about this domain name, we can know this domain is still alive.

Domain names are organized in subordinate levels (subdomains) of the DNS root domain. In general, domain names assigned to organization company or individual are second-level domain (2LD) or third-level domain (3LD) in actual use. In our research, domain names refer to second-level domain and below. Some particular domains such as com.cn, for having no actual significance, are handled as top-level domain (TLD).

The domain names generated by DGAs have a certain generating, organizing and utilizing way. To ensure that every bot in Botnet can connect to one C&C at one time, DGAs often use time or hot topics as seeds[5]. So this can make sure all bot can visit the same domain name at the same time. In this case, based on query traffic from RDNS, we can see that, in a certain period of time, a DGA-domain is queried by a group of clients, but in next period of time, this domain has no query at all.

In the aspect of DGA-domain organization, some botnet often utilize one or several 2LD(may be 3LD) to generate lots of subdomains for later use. So in our research, we need to cluster these domains into one group.

In addition to improve the analysis cost, hiding the domain names which are truly used for confusing purpose, the main task of DGA-domain is pathways as internal contact. When these domains are queried, there will show some characteristics, distinct them from legal domains. The figure below showsf the query characteristic between DGA-domain and legitimate domain.

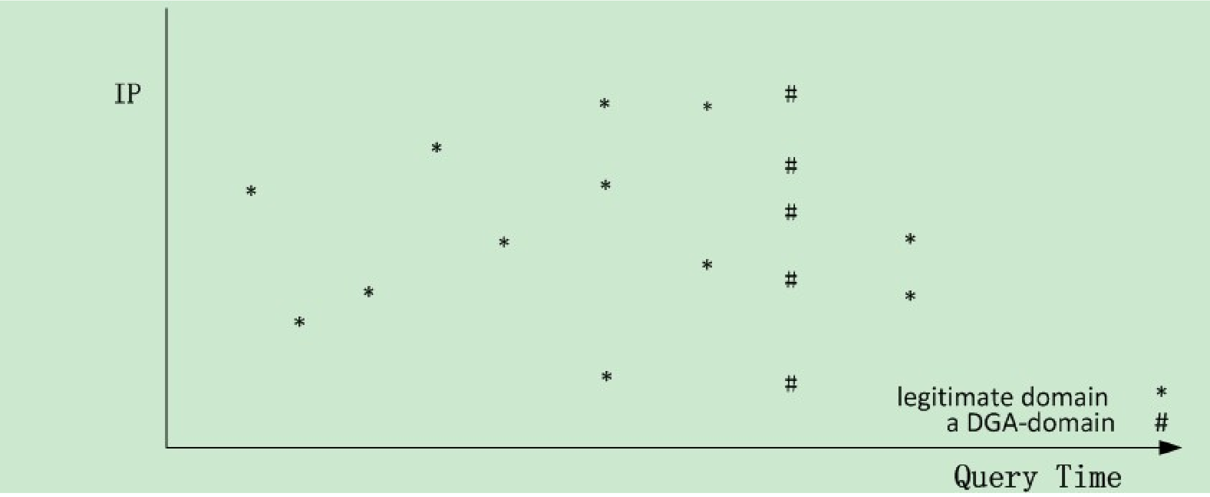

Figure 2: Query characteristic between DGA-domain and legitimate domain

As shown above, the horizontal axis represents domain request time, vertical axis stand for DNS client IP. There is a dot when a domain is queried by one client IP at one time. As time goes on, the request to legal domain appears randomly, and active time of legal domain cross a long period compared to DGA-domain. The DGA-domain is queried only in a short time.

Finger 3 below shows some DGA-domains used in one botnet:

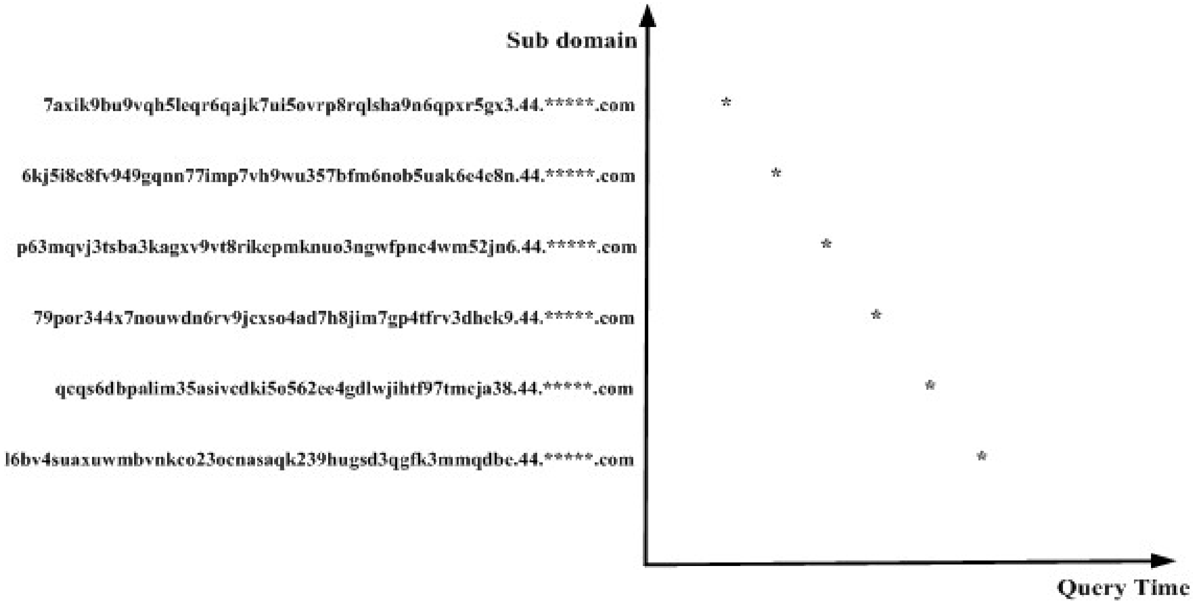

Figure 3: Some DGA-domains used in one botnet

In one botnet, as time goes on, the new domains are generated and used. Every domain has different active time, but they have same life span.

Because all bots in one botnet are controlled by the same C&C, and the whole network command response mechanism of interaction is determined in advance, so the visit of every bot to all DGA-domains have similarity and coherence.

For each DGA-domain in one botnet, there active period only accounted for a short part of entire botnet life time. Although after one period, one DGA-domain can still get resolution, each bot will not use it any more. If we collect and cluster these DGA-domains to one group, we will see these domains point to the same parsed IP group. Dividing all domains into groups according to same second-level, third-level and parsed ip group, all DGA-domains in one group should have the same active life span, and that means there life span should be close to a figure.

For a given time period T, dividing all domain names belong to this period into groups according to same second-level, third-level and parsed ip, we will get domain sets. One domain set is expressed as D=$\{s_1,s_2 \ldots s_n\}$. If domain name s_n is queried at T_1 for the first time, and get last query at T_2, then this domain name's active life is from T_1 to T_2. We express this domain's active life span as , and use count() represents DGA-domain number which belong to one domain set. So for a domain set D, the domain active situation:

$$\{\text{count}(\Delta T_1\ )\text{count}(\Delta T_2\ ) \ldots \text{count}(\Delta T_n\ )\}$$

When the value of count() accounted for the greater proportion, the more distributed domain set is.

Use Distribute(D) represent domain set active time distribution:

$$Distribute(D) = \max(count(\Delta T_i))/\sum_{i=1}^{n} count(\Delta T_i)$$

The greater value of Distribute (D), the more concentrated for the active time length distribution of domain set. The Distribute (D) <=1. Using this feature we can filter out suspected DGA-domains.

5 Performance Analysis

In this section, we present the experimental results of our system.

In consideration of the huge DNS traffic in RDNS, we utilize white list mechanism to reduce traffic flow and improve processing speed. The traffic within white list will not record and handle. We use the top 10000 most popular domain names published by alexa.com to construct white list. According to our statistics, Traffic within the list of accounted for more than 80% of the total flow.

We collect NXDomain traffic from several pilot DNS servers in China. The raw average DNS flow is 150,000 packets per second. DNS response records are merged every five minutes. All data span cross an half of month. Utilizing our method mentioned above, we filter out some domain names. We confirm the domain names filtered using two methods: domain blacklists for cross validation and by tracking other features that related to the domain names. Domain blacklists were published by some companies or security organizations like McAfee site advisor and malware domain list. The disadvantage of this approach is that domain blacklist could not cover all domain names we filtered and have delay before it is published. So for unconfirmed domain names, we tracked other features for further analysis such as the presence of the fast flux, ports on server that domain pointed to, and client address distribution. This step would take several days before the domain names were confirmed.

1. PART OF RESULT DETECTED

Table 1 list part of result detected, when we take the threshold of Distribute (D)as 0.9. After several attempts, we find that when the threshold of Distribute (D)value above 0.9, we can get better detection effect.

Domain list
zscdw.com
xcder.com
cgfde.com
rose7vc.com
ml9fds7.com
quilution2.com

About these domain filtered above, we did some statistics like the number of distinct A records. Table 2 shows the number of parsed IP about one domain at different time.

Warning: TRIAL RESTRICTION – Table omitted!

1. NUMBER OF PARSED IP CONTAINED BY ONE DOMAIN

When taking a closer look at the parsed IPs between every domain, some have a great intersection. So this may indicate that some domains may be used by one malware.

6 Conclusions

In this paper, we propose a methodology to detect DGA-based botnet based on the query characteristic of DGA-domains. Our system is deployed on the pilot RDNS of Internet operators. According to the experiments, our system can filter out some DGA-based botnet successfully.

In previous work, Bayes and SVM are often used to classify domain names. These works need the support of the known data and the classification results are affected by the training data.

Unlike previous work, our approach does not rely on external resources such as known malware domain names. Compared with the method based on the domain alphanumeric characters, this method does not depend on the specific character of domain, and can enhance the effectiveness of existing detection method. Meanwhile, longer time is needed to confirm the suspicious domain list compared to other methods, and this needs to be improved in our further research.

References

[1] M. Antonakakis, R. Perdisci, Y. Nadji, N. Vasiloglou, S. Abu-Nimeh, W. Lee, and D. Dagon. From throw-away traffic to bots: Detecting the rise of dga-based malware. In *Proc. of the 21th USENIX Security Symposium (Security'12), Bellevue, Washington, USA*, pages 48–61. USENIX Association, August 2012.

[2] P. Porras, H. Saidi, and V. Yegneswaran. A foray into conficker's logic and rendezvous points. In *Proc. of the 2nd USENIX Conference on Large-scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and More (LEET'09), Boston, Massachusetts, USA*. USENIX Association, April 2009.

[3] M. Rajab, J. Zarfoss, F. Monrose, and A. Terzis. A multifaceted approach to understanding the botnet phenomenon. In *Proc. of the 6th ACM SIGCOMM Conference on Internet Measurement (IMC'06), Rio de Janeiro, Brazil*, pages 41–52. ACM, October 2006.

[4] P. Royal. Analysis of the kraken botnet. `http://www.damballa.com/downloads/pubs/KrakenWhitepaper.pdf`, April 2008.

[5] P. Royal. On the kraken and bobax botnets. `htttp://www.damballa.com/downloads/r_pubs/kraken_Response.pdf`, April 2008.

[6] B. Stone-Gross, M. Cova, L. Cavallaro, B. Gilbert, M. Szydlowski, R. Kemmerer, C. Kruegel, and G. Vigna. Your botnet is my botnet: analysis of a botnet takeover. In *Proc. of the 16th ACM Conference on Computer and Communications Security (ACM CCS'09), Chicago, Illinois, USA*, pages 635–647. ACM, November 2009.

[7] S. Yadav, A. Reddy, A. Reddy, and S. Ranja. Detecting algorithmically generated malicious domain names. In *Proc. of the 10th ACM SIGCOMM Conference on Internet Measurement (IMC'10), Melbourne, Australia*, pages 48–61. ACM, November 2010.

Author Biography

Yonglin Zhou is a senior engineer in the national CNCERT (Computer Emergency Response Team of China). He joined CNCERT in 1999 and began his research on information security. He was the core member of several national-level research projects on Internet security. From 2004, he began to lead the AD Dept., focusing on cyber security monitoring, early warning and responding. By his efforts, CNCERT formed the China National Vulnerability Database (CNVD). He also worked for the Internet Society of China (ISC) as the General Secretary of the Information Security Committee. He is well connected with governments, ISPs, ICPs, CIIPs and cyber security companies. To fight against the malware economy, he formed the Anti Network Virus Alliance (ANVA). Because of his remarkable expertise, Mr. ZHOU has been invited as the Advanced Info. Security Advisor of Beijing Olympics. Mr. ZHOU is very active on international cooperation. He spoke many times at the meetings of APEC-Tel, FIRST, APCERT, etc. He is co-chairing the China-US joint working group on cyber security dialogue, hosted by the Internet Society of China and the East-West Institute.

Qingshan Li received the BE in Management Information System from Northeastern University, the ME in Software Engineering from Peking University, China, in 1999, 2012, respectively. Now he has been completing a PhD in Computer Software and Theory from Peking University. He joined Peking University in October 2009 as an associate research fellow. He served as the technical director vice president of Network Security department at Neusoft Group before 2009. Now as the associate director in Information Security Laboratory of Peking University, his primary research interests involve network security and intelligent mobile terminal securityespecially the detection technology for APT(Advanced Persistent Threat).

Qidi Miao received the BE in Software Engineer fom Suqian College in 2012. He is currently a graduate student in Software College, Northeastern University, China. His primary research interests are MIPv6/HMIPv6 security, wireless mesh network security, Internet security, as well as security and privacy in ubiquitous computing.

Kangbin Yim received his B.S., M.S., and Ph.D. from Ajou University, Suwon, Korea in 1992, 1994 and 2001, respectively. He is currently an associate professor in the Department of Information Security Engineering, Soonchunhyang University. He has served as an executive board member of Korea Institute of Information Security and Cryptology, Korean Society for Internet Information and The Institute of Electronics Engineers of Korea. He also has served as a committee chair of the international conferences and workshops and the guest editor of the journals such as JIT, MIS, JISIS and JoWUA. His research interests include vulnerability assessment, code obfuscation, malware analysis, leakage protection, secure hardware, and systems security. Related to these topics, he has worked on more than fifty research projects and published more than a hundred research papers.